

906F9/24

(12) UK Patent Application (19) GB (11) 2 079 996 A

- (21) Application No 8119861
 (22) Date of filing 26 Jun 1981
 (30) Priority data
 (31) 68050
 (32) 3 Jul 1980
 (33) Italy (IT)
 (43) Application published
 27 Jan 1982
 (51) INT CL³
 G06F 9/06
 (52) Domestic classification
 G4A 15X 16D 17B FL
 (56) Documents cited
 GB 2054909A
 GB 2054203A
 GB 1492081
 (58) Field of search
 G3N
 G4A
 (71) Applicant
 Ing C Olivetti & C. SpA,
 Via G Jervis 77, 10015
 Ivrea (Turin), Italy
 (72) Inventors
 Gianni Pavan,
 Mario Lorenzi
 (74) Agents
 Reddie & Grose,
 16 Theobalds Road,
 London WC1X 8PL

(54) Data Processing Apparatus with a Predetermined Read-only Memory

(57) A data processing apparatus 13 is provided with a keyboard 6 and a programmable read-only memory 5, for example of EPROM type, for durably preserving data which define the manner of operation of the apparatus. A programming circuit 9

integrated into or connected to the apparatus 13 allows a new memory 10 of the same type to be programmed either by automatic recording of data contained in the original EPROM memory 5, or by data entered by means of the keyboard 6 in order to make available a new read-only memory 10 provided with data modified relative to the data of the original memory 5.

605B 14/405 18.7.81
 79/403 11.1.81
 76.17.72 = 45 3944984 G05B 14/04C2
 = G13 1492081 2101
 2

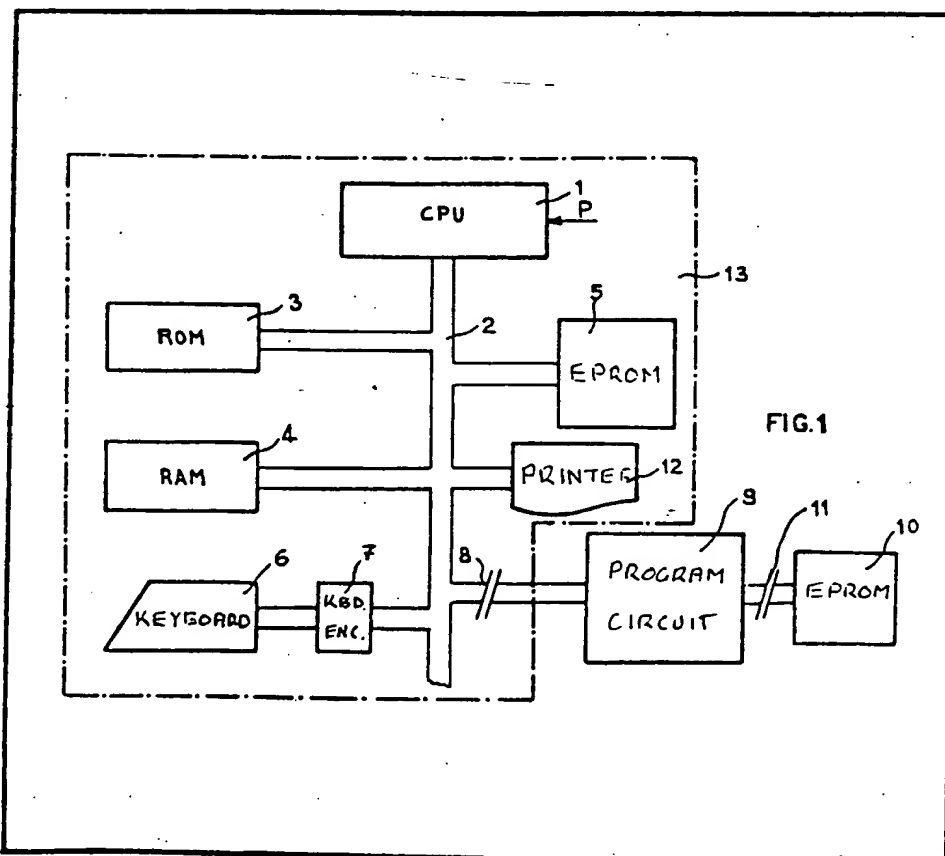
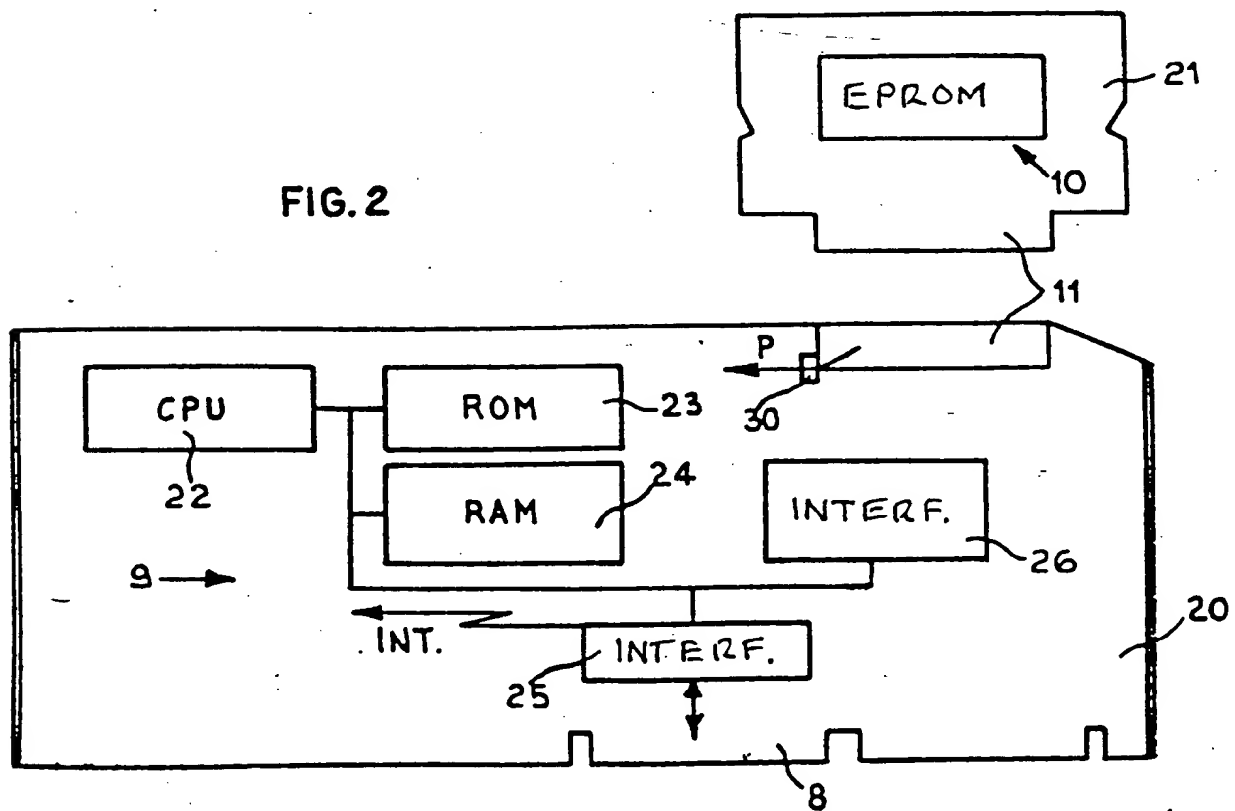
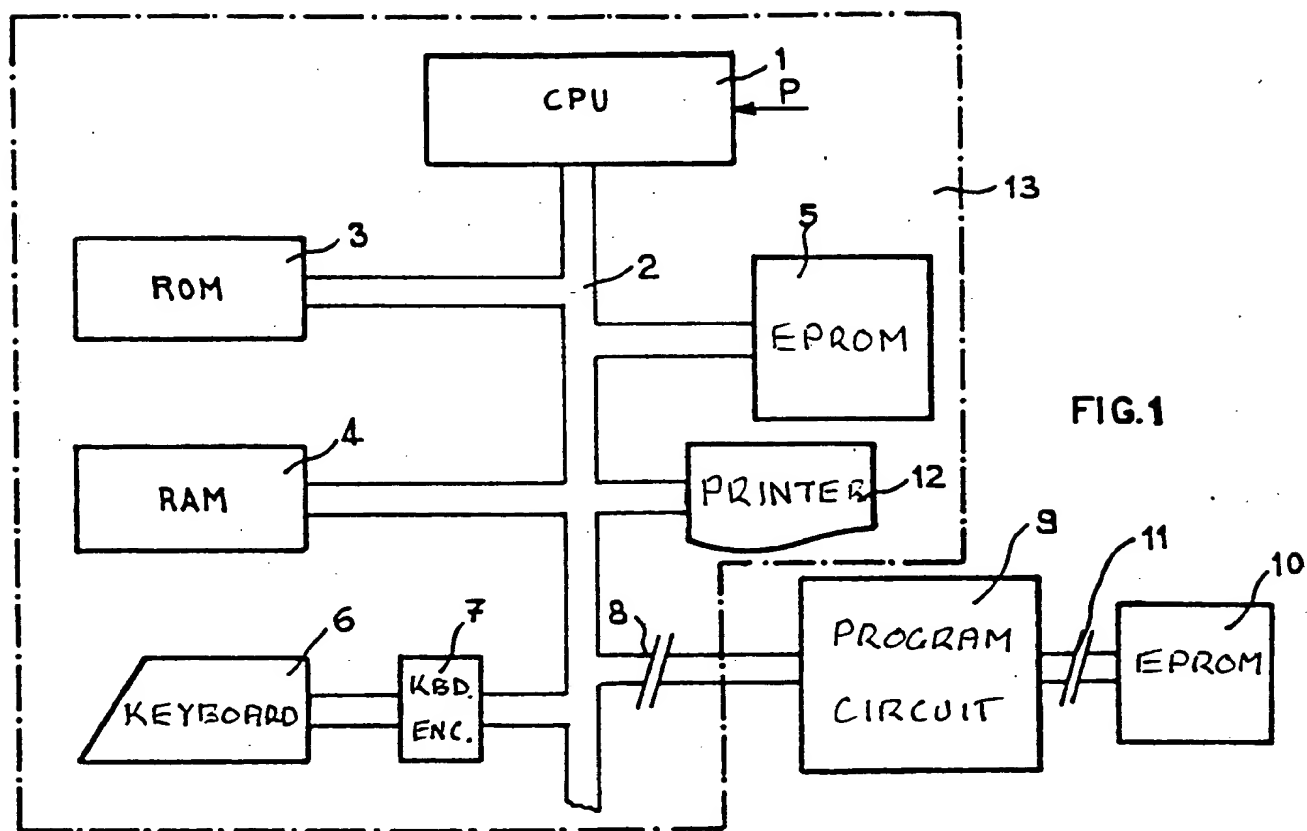


FIG.1

GB 2 079 996 A

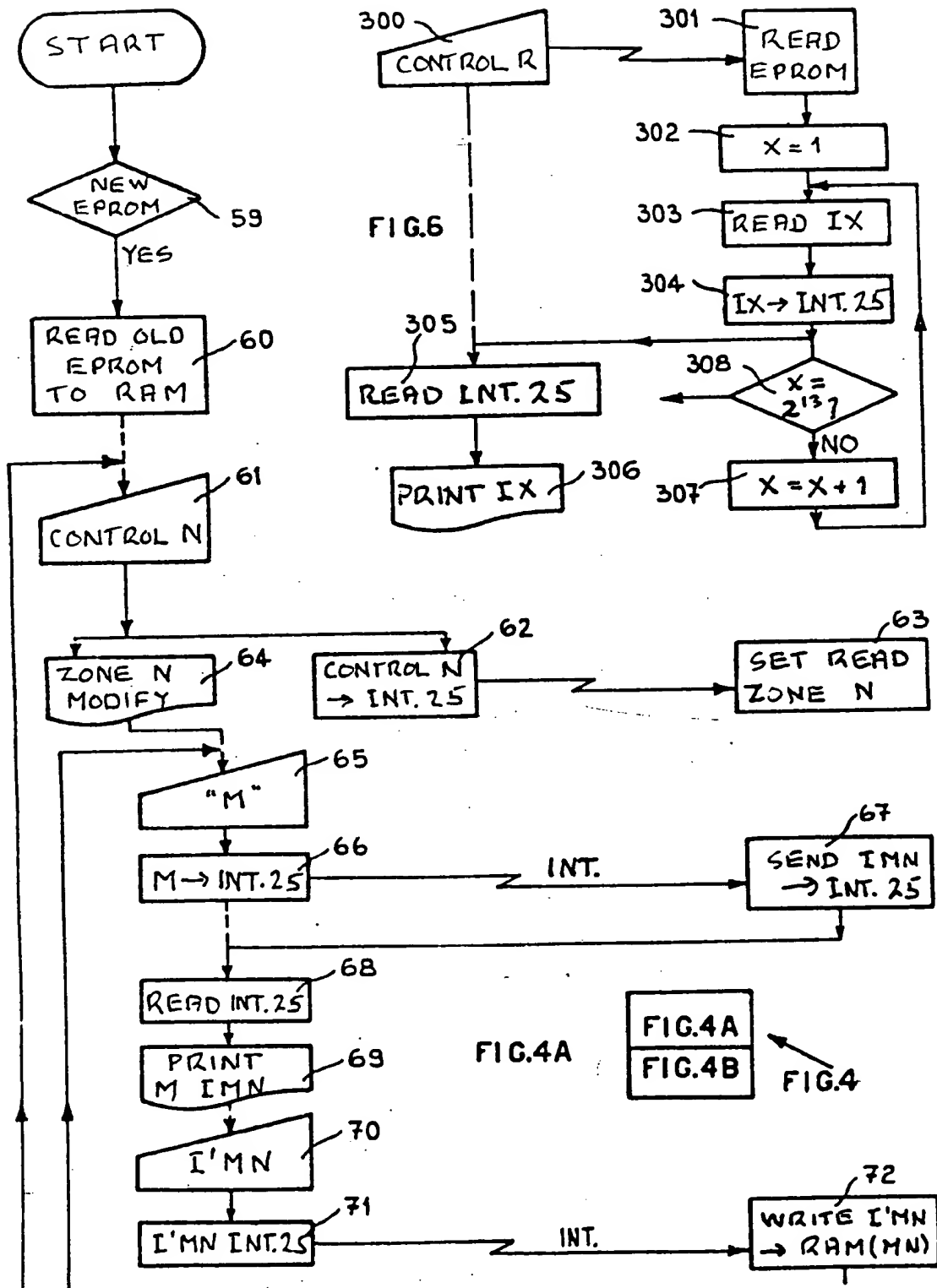
DOC

1/5

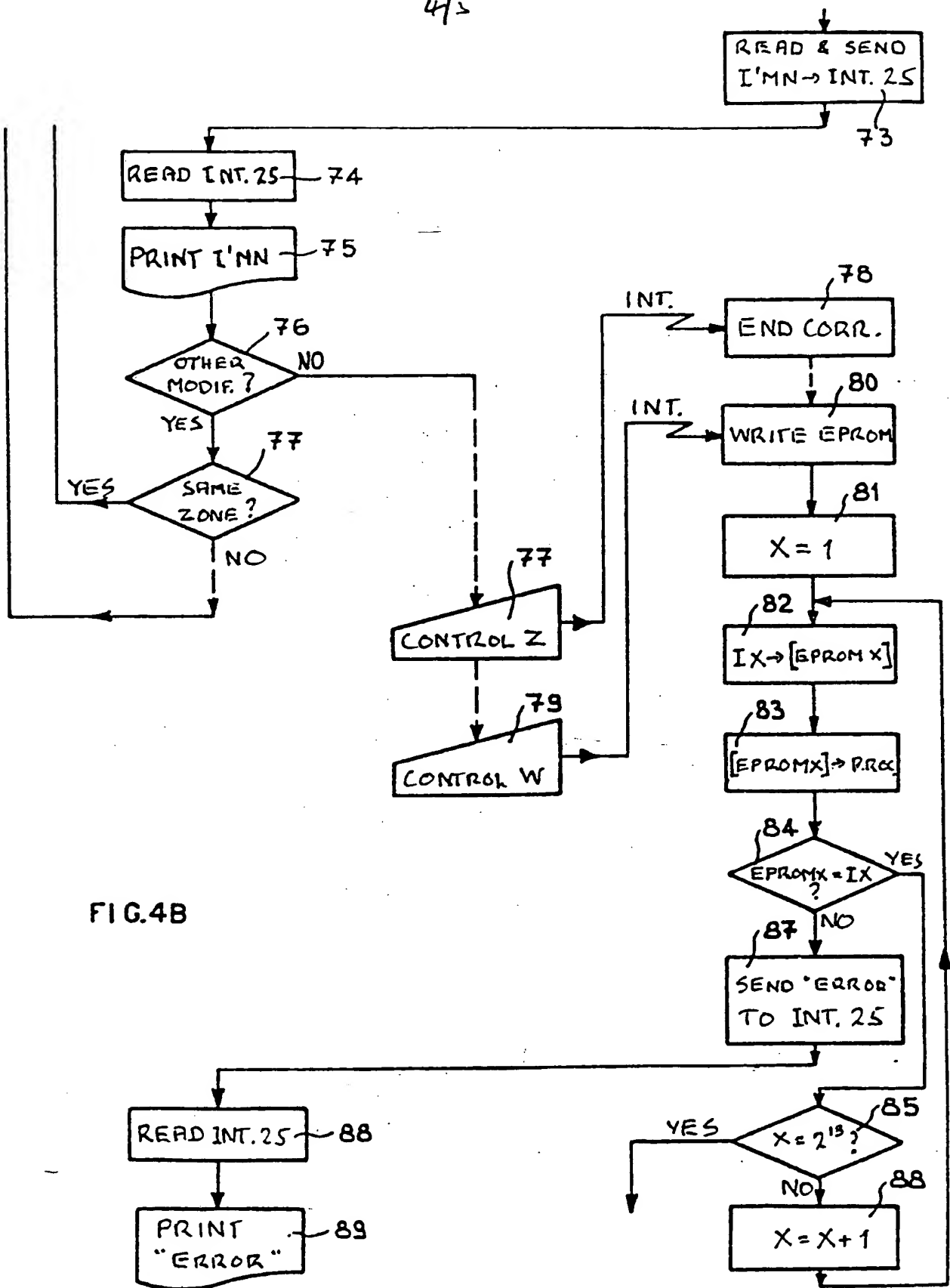




3/5



4/5



5/5

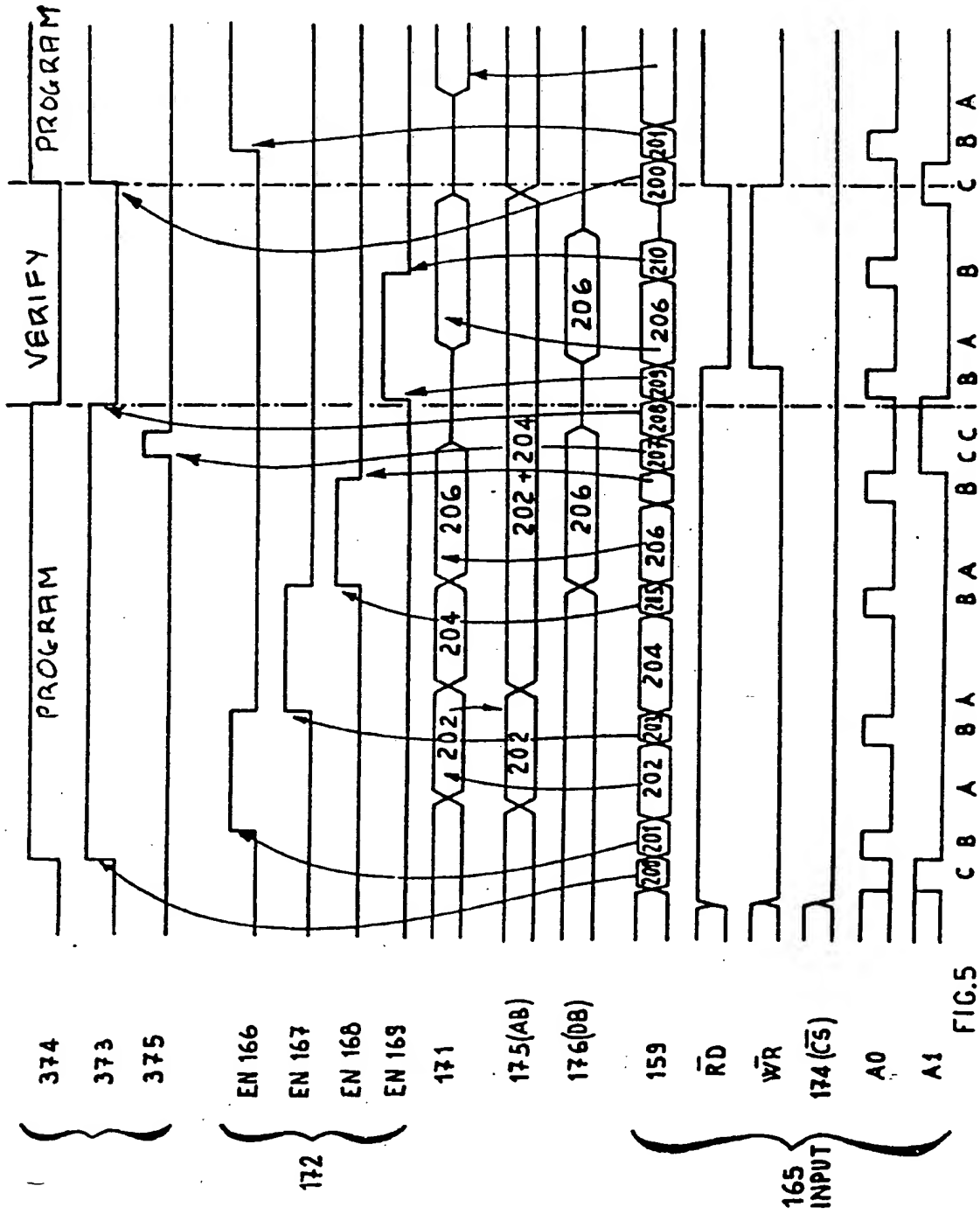


FIG.5

SPECIFICATION **Data Processing Apparatus With a** **Programmable Read-only Memory**

The invention relates to a data processing apparatus with a programmable read-only memory, a central processor, and a keyboard for entering commands or data, in which the memory is arranged to contain data necessary for the operation of the apparatus.

Memories known by the name of PROM (programmable ROM) or EPROM (erasable programmable ROM) are read-only memories, and are programmed in order to durably preserve general data necessary for initializing the machine or for controlling its subsequent operation. PROM and EPROM memories are programmable one or more times respectively, by means of an electrical apparatus.

The advantage of using read-only memories of PROM and EPROM types lies in the facility for programming or updating the permanent data required by the apparatus. The need to update permanent data, which is important in any data processing system, is particularly felt in the case of electronic teleprinters, because such permanent data can determine the method of message communication, the automatic response, the line timing and other variable parameters, according to the local situation in which the machine is used. It is known to update such data by means of a suitable device separate from the apparatus in which the PROM or EPROM memory is used.

The object of the invention is to make updating of the data stored in PROM or EPROM memories more rapid and reliable, and to make the checking thereof more reliable.

The data processing apparatus according to the invention is defined in claim 1 below.

A preferred embodiment is described hereinafter with reference to the accompanying drawings in which:

Figure 1 is a general block diagram of a data processing apparatus embodying the invention;

Figure 2 shows the physical arrangement of the components and connections of a programming circuit 9;

Figure 3 is a diagram of the electrical programming circuit;

Figure 4 (Figs 4A and B assembled) is a flow chart of the successive programming operations;

Figure 5 shows the timing of the main electrical programming signals of the memory;

Figure 6 is a flow chart of the complete printing operations of the EPROM memory.

The data processing apparatus 13 (Figure 1) is constituted by a central processor 1 connected by a bus 2 to an ROM (read only memory) 3, to an RAM (random access memory) 4, to an EPROM permanent memory 5 and to a printer 12.

A keyboard 6 generates decoded commands and data over the bus 2 by means of a keyboard coder 7. A programming circuit 9 is connected to the apparatus 13 by means of a connector 8 in

order to programme a new EPROM 10 connected thereto by means of a connector 11. The apparatus 13 may be an electronic teleprinter station of the type described in our British patent application No. 8118136. In particular, the EPROM 5 corresponds to the EPROM 23 of said patent application, and is arranged to contain at least the automatic response data, the data relative to the characteristics of the station and line, the data necessary for decoding particular sequences of characters, and the editing data for message printing. Some of these data are used by the operating system for initialising the teleprinter on switching-on, in known manner.

It is well known that EPROMs can be programmed, after erasing with ultraviolet rays, by recording at high voltage relative to the normal reading and writing voltage of the apparatus, so that it is apparent the EPROMs can be permanently programmed when under such predetermined erasing and feed voltage conditions.

The programming circuit 9 is assembled on a board 20 (Figure 2) of the teleprinter 13, whereas the new EPROM memory 10 is mounted on a board 21. The connector 11 (PCB edge connector) allows direct connection between the boards 20 and 21. When this connection is made, a contact 30 closes to inform the CPU 1 over a line P that the connection has taken place. The connector 8 is arranged to connect the board 20 to the bus 2 of the teleprinter 13 by means of a multiple cable.

The programming circuit 9 comprises essentially a second processor 22, an ROM 23 arranged to contain the operating micro-programmes for the circuit 9, and an RAM working memory 24 for temporarily storing the data contained in the memories 5, 10. The programming circuit also comprises a transfer interface 25, by means of which the processor 22 converses with the central processor 1, and a recording interface 26 by means of which access is gained to the EPROM 10.

The interface 25 allows transfer of one code at a time representative of a command or character destined for the main processor 1 or for the secondary processor 22. Each time the main processor 1 generates a code destined for the second processor 22, the interface 25 stores it and activates an interrupt signal INT which conditions the processor 22 in known manner for accepting the code in order to effect the appropriate operations.

The interface 25 (Figure 3) by means of which conversation is carried on between the secondary processor 22 and main processor 1 is analogous to the interface 50 described in the aforesaid patent application and disposed between the central units 20 and 52 of the basic machine 1 and additional machine 4 of that application respectively. Only the essential principles of operation are summarised hereinafter.

The CPU 1 converses with the interface 25 by way of the bus 2 which comprises a data bus

154, an address bus 150 and some READ and WRITE command lines 357, 358. The processor 1 generates at its output signals READ or WRITE signals and a code over the address bus 150. The combination of the signals and code interpreted by a decoding circuit 151 and by two logic gates 152 and 153 represents a reading or writing selection for the secondary processor 22. In the case of a writing selection, a register 153 is enabled to store a code fed by the processor 1 over the data bus 154, and simultaneously a flip flop 155 is set to generate the interrupt signal INT at the input to the secondary processor 22. It is apparent that the register 153 can store the data fed into the processor 1 from the keyboard 6 or from the EPROM 5.

In contrast, a reading selection has the effect of enabling a register 156 by means of a signal 353, in order to generate over the data bus 154 a code contained therein and fed by the processor 22. The outputs 357 of a flip flop 157 and 358 of a flip flop 158 are examined periodically by the central processor 1, and, if active, indicate the fact that the register 156 or register 153 respectively contain codes ready for transfer.

A DB data bus 159 connects the two registers 153 and 156 to the processor 22, for code transfer in either direction. In addition, the processor 26 generates by way of an AB address bus 160, two different command words for running the interface 25. These command words are decoded by a decoding circuit 161 in order to generate two types of control signal 162 and 163 respectively. The signal 162 enables the register 156 in order to store the code generated by the processor 22 over the data bus 159, and simultaneously activates the flip flop 157 for the explained function. The signal 163 activated by the processor 22 on each interruption signal INT, enables the register 153 in order to generate over channel 159 the code stored therein, and resets the flip flop 158 to indicate that reading has taken place. The flip flop 155 which has set the signal INT is reset by the processor 22 by means of a reset signal.

The recording interface 26 essentially comprises a programmable input-output circuit 165 of INTEL 8255 type, which receives commands and data from the secondary processor 22. The input-output circuit 165 in its turn coordinates the generation of data, addresses and commands destined for programming the EPROM memory 10. Four registers 166, 167, 168 and 169 are included in the circuit 26 together with a voltage boosting circuit 170 for generating a voltage of about 25V necessary for programming the memory 10.

Various modes of operation of the input-output circuit 165 are described in the appropriate manuals published by INTEL Corporation. In the present embodiment the operation chosen is the so-called "O" type.

The data bus 159 of the processor 22 is used for exchanging commands, addresses and data with the circuit 165. The circuit 165 is connected

to three buses 171, 172 and 173 corresponding to three input-output gates A, B and C included in the circuit 165. The three gates A, B and C are enabled selectively by a pair of signals A0 and A1 originating from the address bus 160 of the processor 22 to store the various types of data transferred between the processor 22 and the registers 166—169. Two RD and WR signals generated by the processor 22 in known manner select the direction of transfer.

The circuit 165 is also enabled by a general enabling signal 174 produced by the processor 22 by means of a suitable command word interpreted by the decoding circuit 161.

The bus 171 is used for transferring addresses or data to the registers 166, 167, 168. The bus 172 is used for communicating enabling commands for the registers 166—169, and the bus 173 for generating suitable enabling signals 373, 374 and 375 for the memory 10 and for the voltage boosting circuit 170.

The voltage boosting circuit 170 receives a feed voltage of 38 V which is present in a terminal 380, and is divided by a resistor 382 and a pair of Zener diodes 383 and 384 in order to supply over a line 385 a stable voltage of 25 V suitable for programming the EPROM memory 10 chosen in the present embodiment.

When the signal 374 goes to high lock level, the output 387 of an inverter 386 biases a transistor 390 by way of two resistors 388 and 389. The transistor 390, which now conducts, passes the voltage of 25 V over a line 391 representing the input to the VPP programming pin of the memory 10, so putting this latter in the condition required for programming.

The operation of the apparatus for programming a new EPROM 10 is described hereinafter with reference to Figures 3, 4 and 5. Normally, the programming circuit 9 has no board 21 connected thereto. When the operator needs to programme a new EPROM 10, for example in order to connect the machine to a different telecommunication network, to change the answerback code sent in automatic response for any reason, or for reason of other standard transmission or editing instructions, he has generally to replace certain data of the EPROM 5 with new data, or at least add the new data to the data of the EPROM 5. The operator sets up the new data on the keyboard, while the data to remain unchanged are transferred to the new EPROM 10 from the existing EPROM 5.

The operator firstly connected the new EPROM 10, plugged in a socket on the board 21, to the connector 11 of the board 20 of the programming circuit 9. The operator then switches on the machine, represented by START (Figure 4), after which normal diagnosis operations are carried out followed by a check 59 for the presence of the board 21, which is indicated by a signal originating from the contact 30 (Figure 2) over the line F. If the result of the check is positive, a routine 60 is run (Figure 4a) in which, under the control of the CPU1, the EPROM 5 is read

character by character by way of the interface 25 (Figure 3), and is transferred into the RAM 24 of the circuit 9 under the control of the CPU in a manner analogous to the transfers described in the aforesaid patent application.

It is apparent that at this point the working memory 24 is programmed exactly as the EPROM memory 5, so that each access to the memory 24 identifies the same data contained in the corresponding EPROM address, and the operations necessary for modifying the data are then executed on the memory 24. In order to carry out this modification, the operator executes operation 61 which consists of entering the word "CONTROL" on the keyboard 6 followed by a number indicated generally by "N", which indicates the zone N of the EPROM memory which is required to modify.

The presence of codes relative to the word "CONTROL" at the output of the keyboard decoder 7 (Fig 1) generally conditions the central processor 1 to feed commands or data to the programming circuit 9. In particular, the code "CONTROL N" composed by the operation 61 conditions the main processor 1 to execute two operations 62 and 64. By means of the operation 62 the command "CONTROL N" is stored in the interface 25, and the secondary processor 22, being interrupted, is set (block 63) for reading in the zone N of the working memory 24. By means of the operation 64, the main processor 1 causes the printer 12 (Figure 1) to print the wording: "ZONE N PRESENT MODIFY".

The printer 12 receives the various codes indicative of commands or characters to be printed from the processor 1 over the channel 2. The circuits for decoding the codes received for controlling the various printing operations are located on the printer 12, and are of known type. The operator then introduces from the keyboard 6 a second number "M" (operation 65 in Figure 3) which defines within the zone N the Mth data item which it is required to modify and which will be indicated briefly as IMN.

The values M and N can clearly represent overall either the address of the prechosen data item directly, or a symbolic address, which is transformed by the processor 22 by conventional methods into an effective address, without for this reason changing the concepts herein described.

The response to feeding the number M to the processor 22 (Figure 2) is now the reading of the data item IMN from the memory 24 and its storing 67 (Figure 4) in the interface 25. The central processor 1 then accepts the data item IMN (block 68) to control a second printing operation 69. In particular, the printer 12 is now controlled in the described manner in order to print the data item IMN with the relative position number M in the following arrangement with respect to the preceding writing:

ZONE N	PRESENT	MODIFY
M	IMN	

By means of the keyboard 6, the operator now sets the required modifications (operation 70) or the new I'MN characters to replace those of the original IMN data item which has just been printed.

Summarising, the special cooperation, controlled by the processors 1 and 22, between the programming circuit 8, the keyboard 6 and printer 12 aids the operator in the following ways:

1) By clearly printing the alphanumeric characters originally recorded on the EPROM memory 5 in the form of codes. This is possible because the printer 12 itself of the processing system 13 is used, and which by its nature is arranged to recognise the codes in use.

2) By making it possible to feed-in directly the characters constituting the new data item I'MN by means of the keyboard 6, because the keyboard coder 7 generates the corresponding codes over channel 2, suitable for treatment by the various devices of the processing system 13 and of the programming circuit 9.

3) By clearly printing the new characters fed in. The operation described under points 2) and 3) are described in detail hereinafter.

The new data item I'MN is store (operation 71) in the interface 25. The interrupt INT which follows this conditions the secondary processor 22 to store (block 72) the new data item I'MN at the address MN of the working memory 24, in place of the original data item I'MN. After this, the processor 22 reads the data item I'MN which has just been stored, and records it in the interface 25 (operation 73), from which it will read (operation 74) and printed under the control of the central processor 1 (operation 75).

The printing of the new data item I'MN completes the previously printed writing in the following manner:

ZONE N	PRESENT	MODIFY
N	IMN	I'MN

In this manner, the operator checks the address MN of the modified information item, the original information item IMN and the newly introduced data item I'MN. The fact that the data item I'MN is printed (operation 75) after it has been read (operation 73) from the working memory 24 rather than at the time of its introduction (operation 70) assures the operator that it has been correctly recorded in the memory 24.

If the result of a selection 76 is positive, this indicates that the operator wishes to modify further data. In this case, the procedure follows one of two paths, by virtue of the selection 77, according to whether the modification still relates to the zone N of the memory 22 previously selected or another zone. In the first case, the operations restart from the block 65, which indicates the selection of the Mth data item within the zone N, but clearly with a value of M which is different from the preceding. In the second case however, the operations restart from the block 61

with which a number N is selected indicative of a zone different from that previously concerned.

When all modifications have been made (negative outcome of the selection 76), the operator again enters the command "CONTROL" for the programming circuit 9, followed by the letter "Z", to indicate the processor 22 (blocks 77—78) the end of introduction of the data items MN. Finally, by operations 79, the operator enters a command "CONTROL W" which is interpreted by the secondary processor 22 as a programming command for the new EPROM memory 10 (block 80).

Programming of the EPROM memory 10 begins with a first operation controlled by the processor 22 (Figure 5) for the writing selection of the circuit 165 by means of a high logic level of the signal \overline{RD} and a low level of the signals WR and 174 at the input of the circuit 165.

Immediately afterwards, the signals A_0 and A_1 are brought by the processor 22 to the levels "0, 1" so as to select, as indicated in the last line of Figure 5, the gate C of the circuit 165. At the same time, the processor 22 generates over the data channel 159 a command word 200 which is latched in the gate C and has the effect of enabling the signals 373 and 374 pertaining to the bus 173. The signals 373 and 374 respectively generate a high logic level at the CS pin of the EPROM 10, and a voltage of 25V, obtained in the described manner, at the VPP pin. The memory 10 thus becomes selected for programming operation (operation 80 in Figure 4). The processor 22 then feeds over the data bus 159 a command word 201 (Figure 5) which, accompanied by a "1, 0" configuration of the signals A_0 , A_1 , is latched in the gate B of the circuit 165 with the effect of enabling the signal EN 166 pertaining to the bus 172.

A word 202, then generated by the processor 22 over the bus 159, represents the most significant part of the selected address. It is accompanied by levels A_0 , A_1 which select the gate A, so that the word 202 is immediately latched by way of the channel 171 in the register 166 (Figure 3), which had been previously enabled and of which the output, over bus 175, is represented in Figure 5.

A word 203 addressed to the gate B then enables the register 167 by means of a signal EN 167, and the less significant part 204 of the selected address is stored by the procedure just described, firstly in the gate B then in the register 167.

At this point, the channel 175 presents the complete address 202+204 at the input to the memory 10 of Figure 3. Initially, this address is $X=1$, and the relative operation is indicated by block 81 in Figure 4B. By means of an analogous procedure, the processor 22 enables the register 168 by means of a word 205 (Figure 5) (command EN 168), and causes the storing in this register of a word 206 is obviously taken from the previously programmed working memory 24. It follows that the data item 206 is also located at

the input to the memory 10 by way of the bus 176.

At this point, the processor 22 generates a command word 207 addressed to the gate C of the circuit 165 (configuration "0, 1" of the signals A_0 , A_1). The word 207 enables the input signal 375 to the PD/PGM pin, which times the programming on the memory 10, of the data item over bus 176 at the address presented over bus 175.

At this point, the operation of programming the memory 10 by means of the data item 206 at the address 202+204 has been carried out (operation 82 in Figure 4), and a checking operation follows consisting of re-reading the data item 206 which has just been recorded. The processor 22 now stores a further command word 208 in the gate C of the circuit 165, which brings all three signals 373, 374 and 375 (Figure 3) of channel 173 to low logic level, to enable the memory 10 for a reading operation.

The addressed word generated over channel 176 is the same as the data item 206 which has just been programmed, because the same address 202+204 (Figure 4) remains over bus 175 which addressed the memory 10. At the same time and in the usual manner a word 209 enables the command EN 169 for enabling the register 169 (Figure 3). Consequently, the word 206 is stored in the register 169 and fed over bus 171 as can be seen in Figure 5. With the switching-over of both signals \overline{RD} and WR and the selection of levels "0, 0" of the signals A_0 and A_1 , the processor 22 now causes reading at the input of gate A (operation 83 in Figure 4) of the data item 206 present by way of the bus 171. Having read the data item, the processor 22 then makes a verification test 84 consisting of an operation in which the data item re-read by the EPROM 10 is compared with the corresponding data item IX taken from the memory 24.

If this verification is positive, after a word 210 (Figure 5) necessary for disabling the signal EN 169, the processor 22 again generates the words 200 and 201 in order to prepare the circuit 26 for programming the next data item on the memory 10, and the same operations as heretofore described are then repeated. In particular, if the comparison 84 of Figure 4 gives a positive result, this group of operations is repeated by incrementing the address X (block 86). If however the comparison 84 gives a negative result, the processor 22 passes an "ERROR" code to the interface 25 (block 87). When the central processor 1 examines the interface 25, it takes the error signal by means of the reading operation 88, and prints it in plain language (block 89) by means of the printer 12 (Figure 1). In this manner the operator is always informed of a programming error of the EPROM memory 10.

If no error signal is printed, the operations 82—88 are repeated until a positive result of the selection 85 indicates that transfer of the last data item has been made. At this point the proposed result has been attained, in that the new

EPROM memory 10 is available, programmed as the old EPROM memory 5 but with the required modifications made. The EPROM 5 can be removed from its socket and be replaced by the new EPROM 10.

With reference to Figure 6, a description will now be given of the sequence of operations controlled by the operator by means of a command known as "CONTROL R" typed on the keyboard 6 in order to determine reading and total printing of the EPROM 10. Printing the entire contents of the EPROM 10 enables a complete check to be made of it, and it is useful in particular in order to check if a memory is programmable, i.e. is not already recorded because of which all the read data are at logic level 1.

The type of conversation between the central processor 1 and secondary processor 22 is entirely similar to that described with reference to Figure 4. When the operator types the codes "CONTROL R" (block 300), the secondary processor 22 interprets them as a total reading command 301 for the EPROM 10.

The secondary processor 22 then executes the reading of the first data item IX at the address X=1 (blocks 302 and 303) of the EPROM memory 10 in the same manner as described with reference to the diagrams of Figure 5. By means of operation 304, the processor 22 memorises the data item IX just read on the interface 25, which is read at the due time by the central processor 1 (operation 305) and printed (operation 306).

These operations are repeated starting from block 303 incrementing the address (block 307) each time, until the selection 308 indicates that the entire EPROM memory 10 has been read and printed.

It is apparent that various modifications and improvements can be made to the described apparatus without leaving the scope of the invention as claimed. For example, the circuit 9 can be connected to the apparatus 13 by operating a manual control, or by directly connecting the connector 8 to the apparatus 13. The presence of the plate 21 can be sensed by means of a routine activated at each initialisation of the apparatus, instead of by the contact 30. In addition, the selection of the data item to be modified on the new EPROM 10 can be made by means of a single addressing operation following the CONTROL command, and the data can be checked by a display instead of by means of the printer 12.

Although a special programming socket 11 has been described for the new EPROM it would be possible to modify data paths so that the old EPROM 5 would be read into RAM, and removed from its socket to be replaced by the new EPROM 10. After updating the data in RAM, this would be written into the new EPROM.

Claims

1. Data processing apparatus comprising a central processor and a keyboard for entering

commands and data, a programmable read-only memory which contains operating routines for the apparatus, and a programming circuit which is integrated into the apparatus and is arranged for control by the central processor, the programming circuit being adapted, in a selected condition of the apparatus, to effect programming of a read-only memory in response to data entered from the keyboard.

2. Apparatus as claimed in claim 1, wherein the memory is programmed with data for initialising the apparatus.

3. Apparatus as claimed in claim 1 or 2, for an electronic teleprinter station, wherein the memory is programmed at least with the automatic response data of the station, with data indicative of characteristics of the station and line, and with the character sequence decoding data.

4. Apparatus as claimed in claim 3, wherein the memory is programmed with editing data for printing messages.

5. Apparatus as claimed in any of the preceding claims, wherein the programming circuit is adapted to read data from an only read-only memory, to supplement and/or modify that data in accordance with the data entered from the keyboard, and to write the supplemented/modified data into a new read-only memory.

6. Apparatus as claimed in claim 5, comprising a printer and control means included in the main processor for conditioning the printer to print the data already programmed in the old memory and the data to be programmed by way of the programming circuit.

7. Apparatus as claimed in claim 5 or 6, comprising working memory included in the programming circuit and controlled by the central processor in order to temporarily store the data contained in the old memory and to insert the data introduced by the keyboard.

8. Apparatus as claimed in claim 7, comprising a second processor included in the programming circuit and by a plurality of keys of the keyboard arranged to generate commands for selecting predetermined data stored in the working memory, the central processor being arranged to communicate the selection commands and the data introduced by the keyboard to the second processor.

9. Apparatus as claimed in claim 8, wherein the programming circuit transfer means controlled by the selection commands in order to determine the transfer of the predetermined data to the main processor, the transfer means being further controlled by the second processor in order to determine the transfer of the data contained in the working memory to the main processor for the purpose of executing the printing of the predetermined data and of the data introduced by the keyboard.

10. Apparatus as claimed in claim 7, 8 or 9, comprising manual setting means included in the keyboard and arranged to generate a recording command, the first central processor being

arranged to communicate the recording command to the second processor in order to control a recording interface included in the programming circuit for recording the new memory with the data contained in the working memory.

11. Apparatus as claimed in claim 10, further comprising checking means activated by the second processor each time the programming means programme the new memory with an item of data contained in the working memory, the checking means comprising means for reading the programmed data, means for comparing the programmed data item with the data item contained in the working memory, and means for indicating an error if the comparison is negative.

12. Apparatus as claimed in claim 11, comprising manual control means for the printing operation arranged to cause the reading means to read each data item contained in the new memory and to transfer it to the main processor for controlling the printing of the contents of the new memory.

13. Apparatus as claimed in any of claims 8 to 12; wherein the commands or data introduced by the keyboard activate a selection command for causing the transfer means to temporarily memorise the commands or data and for producing an interrupt signal which conditions the second processor to examine the commands or data.

14. Apparatus as claimed in claim 13, wherein

the transfer means comprise means for storing data generated by the second processor, the main processor periodically examining these means in order to accept the data.

15. A method of operating a data processing system comprising a central processor, a keyboard for typing commands and data as input to the processor and a first read-only memory of the type which can be programmed electrically only under predetermined conditions and is arranged to store data which is permanently available to the system, comprising the following steps:

connecting to the system a programming circuit for the said type of memories, temporarily storing in the programming circuit the data contained in the first memory, modifying the temporarily stored data by means of commands and data typed on the keyboard, connecting to the programming circuit a second memory of the said type and under the said predetermined conditions, programming the second memory with the modified temporarily memorised data, and replacing the first memory with the second memory.

16. Data processing apparatus substantially as hereinbefore described with reference to and as illustrated in the accompanying drawings.

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☒ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.